p\_list from DB and API - design

# overview

today all the config files for our clients are saved in S3 buckets. so every environment has a few files and a different directory, this causes an over head in the creating the files and supporting them. the goal of the feature is to move the files into the DB so the can be managed from central place and to enable easy migrations between environments.

# high level architecture



# 

# API’s

## name : versionControl (this is an old api that must be supported)

Description : this api will tell the clients if they need to update / must update and what is the config file version, according to the config version the clients will know if they need to get a newer version for the configuration file

Type : Get

Secured : false

cloud : yes

request parameters :

* type - application type
* build\_version - the configuration version

response parameters :

{

"available\_update":false,

"must\_upate":false,

"config\_ver":0,

"er":-1,

"erMessage":""

}

## name : GetConfig

Description : this api will return the configuration for the current app and build version

Type : Get

Secured : false

cloud : yes

request parameters :

* appType - application type
* build\_ver
* config\_ver - based on the the return value the return from the version control api

response parameters :

{

* "config\_ver":0,
* "config":{
  + "UseImageResizerForShoppingListEmail":false,
  + "environment":"p",
  + "models":{
    - "changed\_models":"http://cf-api.homestyler.com/hsmweb/api/product/getchanged/?t={{TIME\_STAMP}}",
    - "zip\_url":"https://d3t5p29y393euf.cloudfront.net/i/{{ID}}/model.dat",
    - "category\_url":"http://hsm-prod-assets.s3.amazonaws.com/Categories/{{ID}}.png",
    - "categoryRetina\_url":false
  + },
  + "plist\_fallbacks":[
    - {
      * "utype":4,
      * "url":"https://s3.amazonaws.com/hsm-prod-modellers-assets/ConfigFiles/Ver\_1/1.3.0/modellers\_services\_v1.3.0.plist"
    - },
    - {
      * "url":"https://s3.amazonaws.com/hsm-prod-modellers-assets/ConfigFiles/Ver\_1/1.3.0/modellers\_services\_v1.3.0.plist",
      * "utype":999
    - }
  + ],
  + "app":{
    - "opening\_image\_timer":15,
    - "opening\_image\_fade\_time":2
  + }
* },
* "er":-1,
* "erMessage":""

}

# DB

## table name : AppVersion

desc : this table will provide details for a build version and the application type

|  |  |  |
| --- | --- | --- |
| AppVersionID |  | PK |
| AppType | application type | foreign key to AppTypes |
| AppVersionBuildID | application build ID | string |
| AppVersionNeedUpdate | does application have an update available | boolean |
| AppVersionMustUpdate | does application need to update | boolean |
| AppVersionAddTime | the time the current version was added to the table | timestamp |
| AppVersionConfigVer | a version stamp that will increment each time that the configuration list has updated. | int |
| AppConfigVersionID | ID for the config list, this is the connection between the wanted list of configuration to the build version and the app type | FK  AppConfigVersions |

# DB

## table name : AppConfigVersions

desc : this table is a logic table that will reference a between a config liks and the build version

|  |  |  |
| --- | --- | --- |
| AppConfigVersionID |  | PK |

## table name : AppConfig

desc : this table will have include the list of parameters in order to create a config file for the client

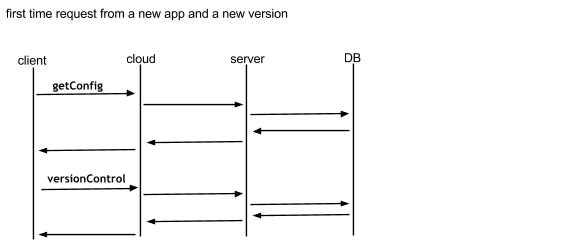
|  |  |  |
| --- | --- | --- |
| AppConfigVersionID | ID for the config list, this is the connection between the wanted list of configuration to the build version and the app type | FK AppConfigVersions,  pk |
| AppConfigID | running id | pk |
| ConfigValueTypeID | a type that will define if the current param is int/ string / array etc | foreign key to ConfigValueTypes |
| AppConfigDesc | the name of the param | string |
| AppConfigValue | the value of the param | string |
| AppConfigParentID | a linq to the parent of the current param, if exists one. | null or key to AppConfigID |

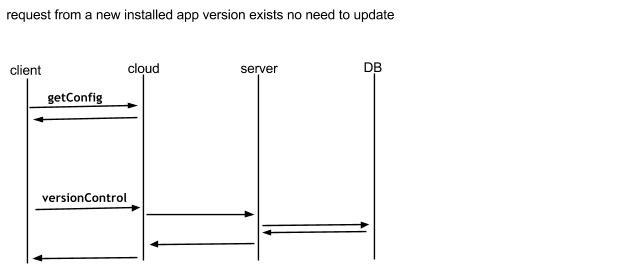
## Table name : ConfigValueTypes

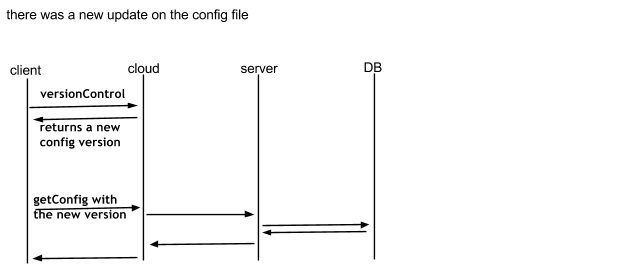
desc : a table that will hold the types of values that can be enterd in the appconfig table

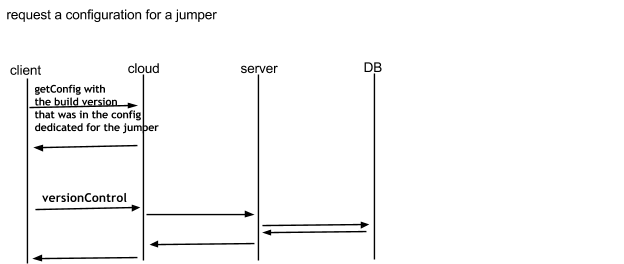
|  |  |
| --- | --- |
| ConfigValueTypeID | int |
| ConfigValueTypeDesc | string |

# sequence diagram









# Guidelines

* In order to add a new config list for a new build

1. add a new key for the version on AppConfigVersions table using -
   1. “insert into AppConfigVersions values ();”
2. add an entry to AppVersion table
   1. AppConfigVersionID - should be set to the to the new key that aws generated on step one
   2. insert into AppVersion (AppType,AppVersionBuildID,AppConfigVersionID ) values (1 or 2 , “build number 1.1.2”, x => the new created key )
3. add the full list of config list to the AppConfig table with the given key creted on step one
   1. insert into AppConfig (AppConfigVersionID, AppConfigID, ConfigValueTypeID,AppConfigDesc,AppConfigValue,AppConfigParentID ) values (x => the new created key , give a key (pk is AppConfigVersionID + AppConfigID), type is from ConfigValueTypes tables, “desc ”, “value is not must”, parent - if not null the set it to the AppConfigID )

* if you update the config list then you need increase the AppVersionConfigVer on AppVersion table
* adding a new build with an existing config list adds a new entry in the AppVersion table and set AppConfigVersionID to have the same value as the previous build.